

# FTML Project

nelson.vicel-farah, antoine.zellmeyer

June 2022

## 1 Bayes estimator and Bayes risk

### Question 1

- Input space  $\mathcal{X} = [0; 1]$
- Output space  $\mathcal{Y} = \mathbb{R}^+$
- $X$  uniform continuous distribution on  $\mathcal{X}$
- $l(x, y) = \text{squared loss}$
- $Y \sim \text{Exp}(1 + X)$

The Bayes estimator in respect to the squared loss is  $f^*(x) = E[Y|X = x]$ , so

$$f^*(x) = E[Y \sim \text{Exp}(1 + x)]$$

We know that  $E[X \sim \text{Exp}(\lambda)] = \frac{1}{\lambda}$  so

$$\mathbf{f}^*(\mathbf{x}) = \frac{\mathbf{1}}{\mathbf{1} + \mathbf{x}}$$

And the Bayes Risk is

$$\begin{aligned} R^* &= E[l(Y, f^*(X))] \\ &= E_X[E_Y[(Y - f^*(X))^2|X]] \\ &= E_X[\text{Var}(Y|X)] \end{aligned}$$

We know that  $\text{Var}[X \sim \text{Exp}(\lambda)] = \frac{1}{\lambda^2}$

$$\begin{aligned} &= E_X\left[\frac{1}{(1 + X)^2}\right] \\ &= \int_0^1 \frac{1}{(1 + x)^2} dx = \left[-\frac{1}{1 + x}\right]_0^1 \\ &= \frac{1}{2} \end{aligned}$$

## Question 2

Let's define  $\tilde{f}$  as the OLS estimator, which means

$$\tilde{f} = \begin{cases} \mathcal{X} \rightarrow \mathcal{Y} \\ x \rightarrow \hat{\theta}_1 x + \hat{\theta}_2 \end{cases}$$

With  $\hat{\theta}_1$  and  $\hat{\theta}_2$  the scalar parameters that minimize the squared loss. We deduce that

$$\begin{aligned} \hat{\theta}_1 &= \frac{Cov(X, Y)}{Var(X)} \\ &= 12 * Cov(X, Y) \\ &= 12 (E[XY] - E[X]E[Y]) \\ &= 12 (E[E[XY|X]] - E[X]E[E[Y|X]]) \\ &= 12 \left( E\left[\frac{X}{1+X}\right] - E[X]E\left[\frac{1}{1+X}\right] \right) \\ &= 12 \left( 1 - \log(2) - \frac{1}{2}\log(2) \right) \\ &= \mathbf{12 - 13\log(2)} \end{aligned}$$

And

$$\begin{aligned} \hat{\theta}_2 &= E[Y] - \hat{\theta}_1 E[X] \\ &= E[E[Y|X]] - \hat{\theta}_1 \frac{1}{2} \\ &= \log(2) - \frac{1}{2}(12 - 13\log(2)) \\ &= \frac{\mathbf{15}}{\mathbf{2}}\log(2) - \mathbf{6} \end{aligned}$$

We conclude that

$$\tilde{f}(x) = (12 - 13\log(2))x + \frac{15}{2}\log(2) - 6$$

The simulation in `part1_simulation.py` with the above settings and 10 000 samples of (X,Y) suggests that the Bayes estimator is better at estimating the setting than the OLS estimator. Probably because our model is not linear. Furthermore, the computed generalization error of the Bayes estimator corresponds to the value of the Bayes Risk we found previously ( $\sim 1/2$ ) which lets us think that the simulation is a good approximation of the theoretical setup.

## 2 Bayes risk with absolute loss

### Question 1

$P(Y|X=x)$  where  $P$  corresponds to an  $\text{Exp}(\lambda)$  continuous distribution.  
The Bayes estimator for  $l_2$  squared loss

$$f_2^*(x) = E[Y|X = x] = E[\lambda e^{-\lambda}] = \frac{1}{\lambda}$$

The Bayes estimator for  $l_1$  absolute loss is the median of  $Y|X=x$  as seen in Question 2

$$f_1^*(x) = \frac{\ln(2)}{\lambda}$$

These Bayes estimators are not equal.

### Question 2

We note  $p = p_{Y|X=x}$

$$\begin{aligned} g(z) &= \int_{\mathbb{R}} |y - z| p(y) dy \\ &= \int_z^{+\infty} (y - z) p(y) dy + \int_{-\infty}^z (z - y) p(y) dy \\ &= \int_z^{+\infty} y p(y) dy - z \int_z^{+\infty} p(y) dy + z \int_{-\infty}^z p(y) dy - \int_{-\infty}^z y p(y) dy \\ \frac{d}{dz} g(z) &= -z p(z) - \left( \int_z^{+\infty} p(y) dy - z p(z) \right) + \left( \int_{-\infty}^z p(y) dy + z p(z) \right) - z p(z) \\ &= \int_{-\infty}^z p(y) dy - \int_z^{+\infty} p(y) dy \end{aligned}$$

Thus,  $\frac{d}{dz} g(z) = 0$  if :

$$\int_{-\infty}^z p(y) dy = \int_z^{+\infty} p(y) dy$$

This occurs when both sides are equal to  $\frac{1}{2}$ , which means that the Bayes estimator  $f^*(x)$  is the median. But to confirm this minimizes  $g(z)$ , let's check the second derivative:

$$\frac{d^2}{dz^2} g(z) = 2p(z) > 0$$

The second derivative is always positive, so our solution is a minimum.

### 3 Expected value of empirical risk

#### Exercise

Step 1 :

$$\begin{aligned}
 E[R_n(\hat{\theta})] &= E\left[\frac{1}{n}\|y - X\hat{\theta}\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|y - X((X^T X)^{-1}X^T y)\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|(I_n - X(X^T X)^{-1}X^T)y\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|(I_n - X(X^T X)^{-1}X^T)(X\theta^* + \epsilon)\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|X\theta^* + \epsilon - \underbrace{X(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{X}}_{=I_d}\theta^* - (X(X^T X)^{-1}X^T)\epsilon\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|X\theta^* + \epsilon - X\theta^* - (X(X^T X)^{-1}X^T)\epsilon\|_2^2\right] \\
 &= E\left[\frac{1}{n}\|\epsilon - (X(X^T X)^{-1}X^T)\epsilon\|_2^2\right] \\
 &= E_\epsilon\left[\frac{1}{n}\|(I_n - X(X^T X)^{-1}X^T)\epsilon\|_2^2\right]
 \end{aligned}$$

Step 2 :

$$A \in \mathbb{R}^{n,m} B \in \mathbb{R}^{n,m}$$

$$tr(A^T B) = \sum_{i,j \in [1,n] \times [1,m]} a_{ij} b_{ij}$$

Thus :

$$\begin{aligned}
 A &\in \mathbb{R}^{n,n} \\
 tr(A^T A) &= \sum_{i,j} a_{ij} a_{ij}
 \end{aligned}$$

**Step 3 :**

$$\begin{aligned} E_{\epsilon} \left[ \frac{1}{n} \|A\epsilon\|^2 \right] &= E_{\epsilon} \left[ \frac{1}{n} \sum_{i=1}^n (A\epsilon)_i^2 \right] = E_{\epsilon} \left[ \frac{1}{n} \epsilon^2 \sum_{i,j \in [1,n]^2} A_{ij}^2 \right] \\ &= E_{\epsilon} \left[ \frac{1}{n} \epsilon^2 \text{tr} (A^T A) \right] \\ &= \frac{1}{n} \text{tr} (A^T A) E_{\epsilon} [\epsilon^2] \\ &= \frac{1}{n} \text{tr} (A^T A) (E_{\epsilon} [\epsilon^2] - \underbrace{\mathbf{E}_{\epsilon} [\epsilon]^2}_{=0}) \\ &= \frac{1}{n} \text{tr} (A^T A) \sigma^2 \end{aligned}$$

Step 4 :

$$\begin{aligned}
A &= I_n - X(X^T X)^{-1} X^T \\
A^T A &= (I_n - X(X^T X)^{-1} X^T)^T (I_n - X(X^T X)^{-1} X^T) \\
&= (\mathbf{I}_n^T - (\mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T))^T (I_n - X(X^T X)^{-1} X^T) \\
&= (\mathbf{I}_n - (\mathbf{X}^T)^T ((\mathbf{X}^T \mathbf{X})^{-1})^T \mathbf{X}^T) (I_n - X(X^T X)^{-1} X^T) \\
&= (\mathbf{I}_n - \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T) (I_n - X(X^T X)^{-1} X^T) \\
&= (I_n - X(X^T X)^{-1} X^T)^2 \\
&= I_n - 2X(X^T X)^{-1} X^T + (X(X^T X)^{-1} X^T)^2 \\
&= I_n - 2X(X^T X)^{-1} X^T + X(X^T X)^{-1} \mathbf{X}^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} X^T \\
&\quad \quad \quad = \mathbf{I}_d \\
&= I_n - 2X(X^T X)^{-1} X^T + X(X^T X)^{-1} X^T \\
&= I_n - X(X^T X)^{-1} X^T = \mathbf{A}
\end{aligned}$$

Thus :  $A^T A = A$

### Step 5 : Conclude

$$\begin{aligned} E[R_n(\hat{\theta})] &= E_\epsilon \left[ \frac{1}{n} \|(I_n - X(X^T X)^{-1} X^T) \epsilon\|_2^2 \right] \\ &= \frac{\sigma^2}{n} \text{tr}(I_n - X(X^T X)^{-1} X^T) \\ &= \frac{\sigma^2}{n} (n - \text{tr}(X(X^T X)^{-1} X^T)) \\ &= \frac{\sigma^2}{n} (n - \text{tr}(X^T X (X^T X)^{-1})) \\ &= \frac{\sigma^2}{n} (n - \text{tr}(I_d)) \\ &= \frac{\sigma^2}{n} (n - d) \end{aligned}$$

Thus :

$$E[R_X(\hat{\theta})] = E[E[R_n(\hat{\theta})]] = E\left[\frac{n-d}{n} \sigma^2\right] = \frac{n-d}{n} \sigma^2$$

## Simulation

### Step 6

$$E \left[ \frac{\|y - X\hat{\theta}\|_2^2}{n-d} \right]$$

We recognize the fixed design risk with a subtle difference :  $\frac{1}{n-d}$  replaced  $\frac{1}{n}$ , so we can use the formula found in Step 5 and apply this change.

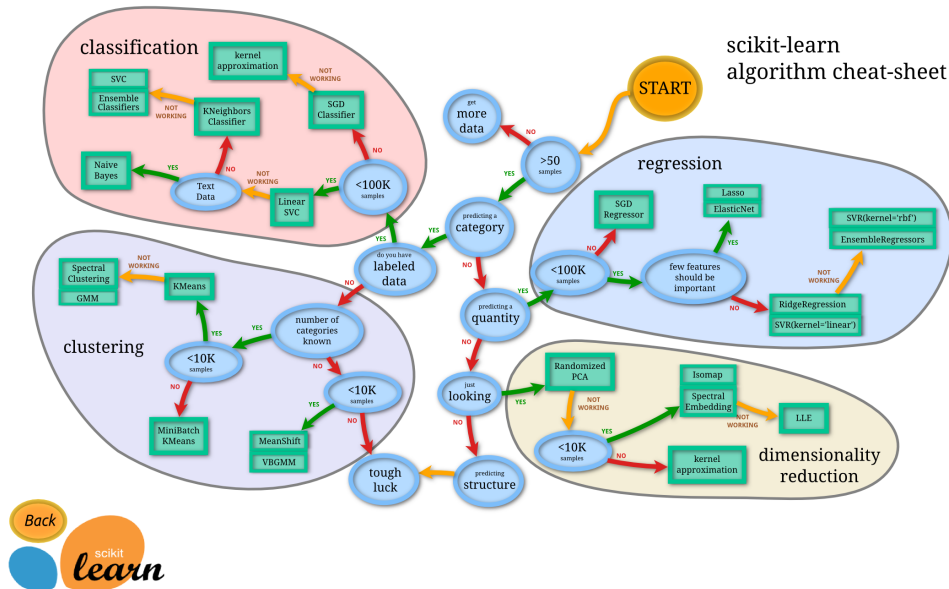
$$= \frac{n-d}{n-d} \sigma^2 = \sigma^2$$

### Step 7

The simulation is in `part3_step7.py`. We suppose :  $n = 1000$ ,  $d = 20$ ,  $y = X\theta^*$  with  $\theta^*$  a random vector. We thus estimate  $\sigma^2$  with the result of Step 6. With this setting, the simulation estimates a variance of 0, which is an expected result knowing that  $y$  is a simple linear function of  $X$  without any noise added to it.

## 4 Regression

For this section and the next one, we chose `scikit-learn` as our machine learning library because it contains many regression and classification models and allows us to follow this decision tree :



We should then retrieve the important properties of the dataset :

Number of entries : 1000

Number of features : 20

This needs a regression model that can handle a small set of samples and 20 features

| Model                       | R2 score       |
|-----------------------------|----------------|
| SVR Linear                  | 0.88757        |
| SVR rbf                     | 0.66461        |
| Ridge alpha=12.0            | <b>0.89021</b> |
| Ridge alpha=15.0            | <b>0.89027</b> |
| Lasso alpha=12.0            | 0.83622        |
| RandomForestRegressor       | 0.82288        |
| AdaBoost Regressor          | 0.82405        |
| Gradient Boosting Regressor | 0.86165        |

Although we should note that the R2 score is highly influenced by the random initialization of the models parameters, the Ridge regression seems to be globally the best regression model in this case.

## 5 Classification

As we did in the previous section, we retrieve useful information about the dataset

Number of entries : 1000

Number of features : 20

This needs a classification model that can handle a small set of samples and 20 features. Let's try some of them:

| Model                      | Accuracy     |
|----------------------------|--------------|
| LinearSVC                  | <b>0.916</b> |
| KNeighbors                 | 0.864        |
| SVC                        | 0.892        |
| RandomForestClassifier     | 0.876        |
| AdaBoostClassifier         | 0.88         |
| GradientBoostingClassifier | 0.868        |

Although we should note that the accuracy is highly influenced by the random initialization of models parameters, the linear SVC seems to be globally the best classification model in this case.